More on Calibration and Reconstruction

Raul Queiroz Feitosa

Objective

This chapter contains additional information on Camera Callibration and Reconstruction.

Contents:

- Recalling Projective Geometry
- Vanishing Points
- Camera Calibration from VPs
- Camera from Fundamental Matrix

Recall Projective Geometry

2 Dimensional

	$\mathbb{P}^2$	$\mathbb{R}^2$
points	$\boldsymbol{x} = (kx_1,kx_2,k)^T = (x_1,x_2,1)^T$ for $k \neq 0$ $\boldsymbol{x}_\infty = (kx_1,kx_2,0)^T = (x_1,x_2,0)^T$ for $k \neq 0$	$\boldsymbol{x} = (x_1,x_2)^T$
lines	$\mathbf{l} = (kl_1,kl_2,k)^T = (l_1,l_2,1)^T$ for $k \neq 0$ $\mathbf{l}_\infty = (0,0,k)^T$ for $k \neq 0$	$\mathbf{l} = (a,b,c)^T$ -

points at infinity

lines at infinity

Recall Projective Geometry

2 Dimensional

- If a point $\boldsymbol{x}$ lies on line $\mathbf{l}$

$$\boldsymbol{x}^T \mathbf{l} = 0$$

- The intersection $\boldsymbol{x}$ of two lines $\mathbf{l}_1$ and $\mathbf{l}_2$

$$\boldsymbol{x} = \mathbf{l}_1 \times \mathbf{l}_2$$

- The line joining points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$

$$\mathbf{l} = \boldsymbol{x}_1 \times \boldsymbol{x}_2$$

- Note that in $\mathbb{P}^2$ parallel lines intersect at infinity

$$(a,b,c) \times (a,b,c') = (\, b(c'-c), a(c-c'), 0)$$

Recall Projective Geometry

3 Dimensional

	$\mathbb{P}^3$	$\mathbb{R}^3$
points	$\boldsymbol{x} = (kx_1, kx_2, kx_3, k)^T = (x_1, x_2, x_3, 1)^T$ for $k \neq 0$ $\boldsymbol{x}_\infty = (kx_1, kx_2, kx_3, 0)^T = (x_1, x_2, x_3, 0)^T$ for $k \neq 0$	$\boldsymbol{x} = (x_1, x_2, x_3)^T$
planes	$\boldsymbol{\pi} = (k\pi_1, k\pi_2, k\pi_3, k)^T = (\pi_1, \pi_2, \pi_3, 1)^T$ for $k \neq 0$ $\boldsymbol{\pi}_\infty = (0,0,0,k)^T$ for $k \neq 0$	$\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3)^T$ -

points at infinity

planes at infinity

Recall Projective Geometry

3 Dimensional

- If a point $\boldsymbol{x}$ lies on plane $\boldsymbol{\pi}$ $\longrightarrow$ $\boldsymbol{x}^T \boldsymbol{\pi} = 0$
- Three (non collinear) points define a plane $\longrightarrow$ $\begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \boldsymbol{x}_3^T \end{bmatrix} \boldsymbol{\pi} = \mathbf{0}$
- Three planes define a point $\longrightarrow$ $\begin{bmatrix} \boldsymbol{\pi}_1^T \\ \boldsymbol{\pi}_2^T \\ \boldsymbol{\pi}_3^T \end{bmatrix} \boldsymbol{x} = \mathbf{0}$
- Note that in $\mathbb{P}^3$ points at infinity lie on a plane at infinity

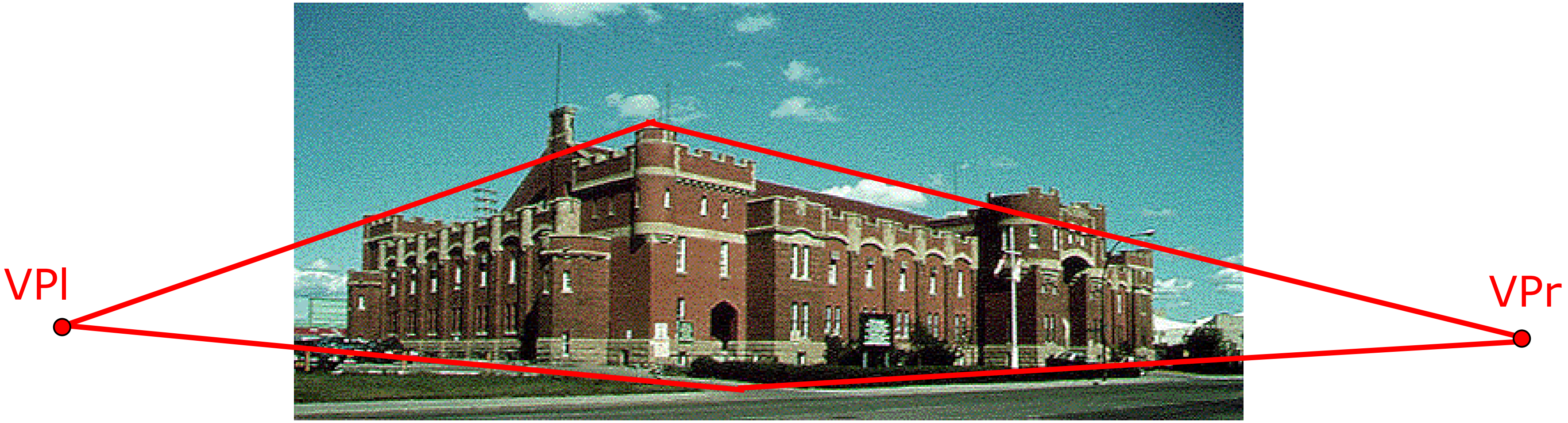
$$(x_1,x_2,x_3,0)^T\,(0,0,0,k) = 0$$

Contents:

- Recalling Projective Geometry
- Vanishing Points
- Camera Calibration from VPs
- Camera from Fundamental Matrix

Vanishing Points

Examples



Vanishing Points

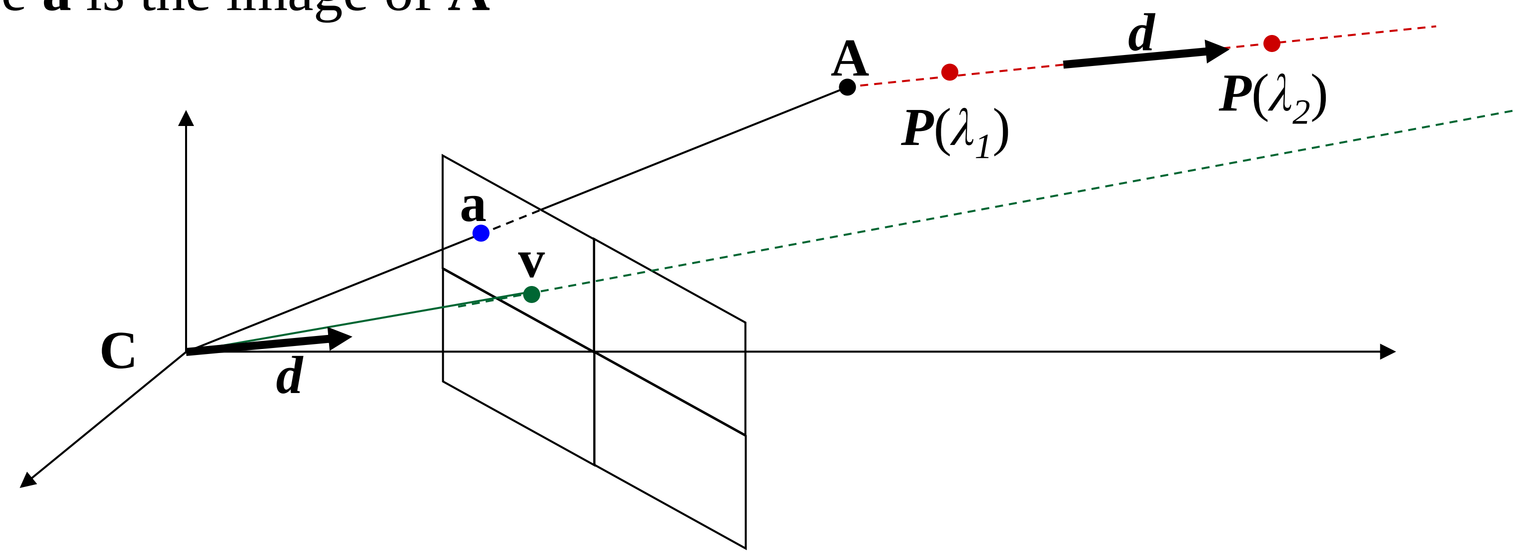
A point in 3D space through point $\boldsymbol{A}=(\mathbf{a}^T,1)^T$ moving in direction $\boldsymbol{D}=(\boldsymbol{d}^T,0)^T$

$$\boldsymbol{P}(\lambda)=\boldsymbol{A}+\lambda\boldsymbol{D}$$

appears on a projective camera $\mathcal{M} = \mathcal{K}\,[\,\mathrm{I}\mid 0]$ in homogeneous coordinates

$$\boldsymbol{p}(\lambda) = \mathcal{M}\ \boldsymbol{P}(\lambda)/z(\lambda) = (\mathcal{M}\boldsymbol{A}+\lambda\mathcal{M}\boldsymbol{D})/z(\lambda) = (\mathbf{a} + \lambda\mathcal{K}\boldsymbol{d})/z(\lambda)$$

where **a** is the image of **A**

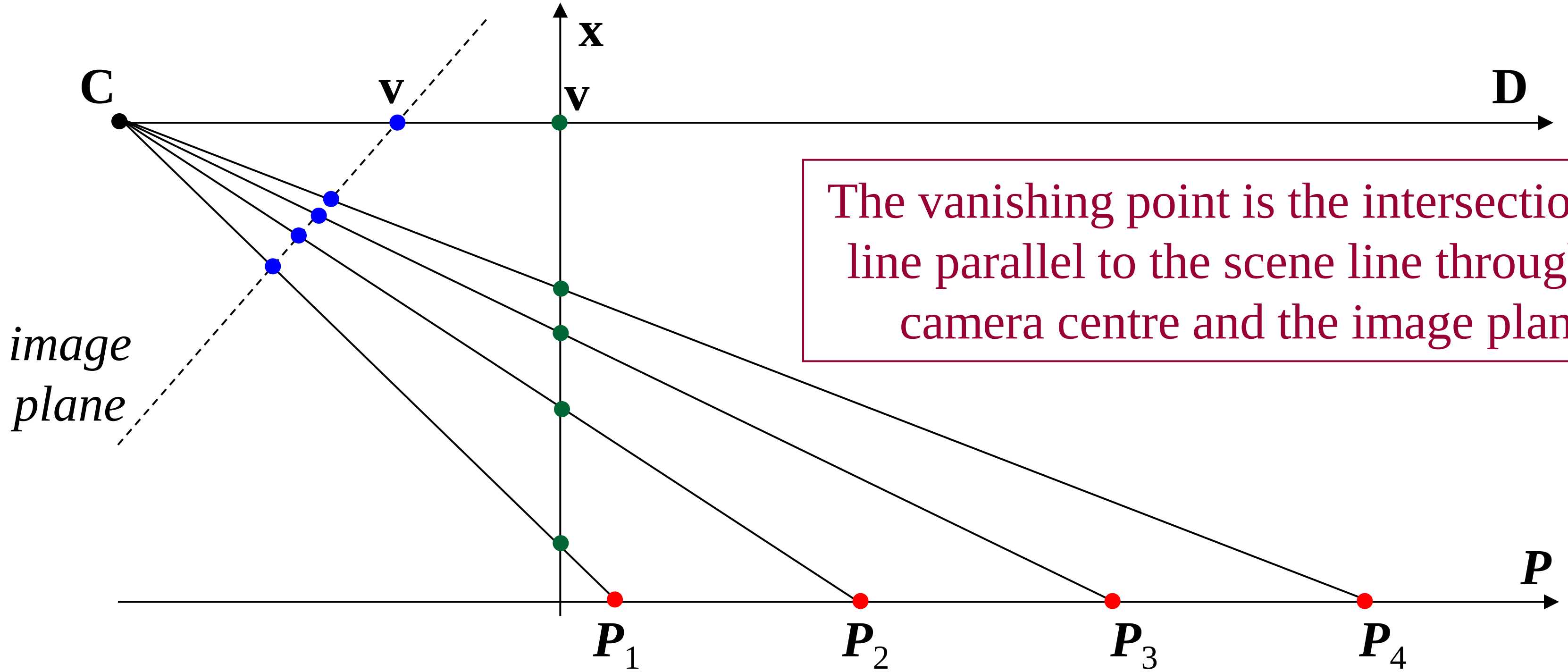


Vanishing Points

The vanishing point is given by

$$\mathbf{v} = \lim_{\lambda\to\infty} \boldsymbol{p}(\lambda) = \lim_{\lambda\to\infty} [(\mathbf{a} + \lambda \mathcal{K}\boldsymbol{d})/z(\lambda)] \propto \mathcal{K}\boldsymbol{d}$$

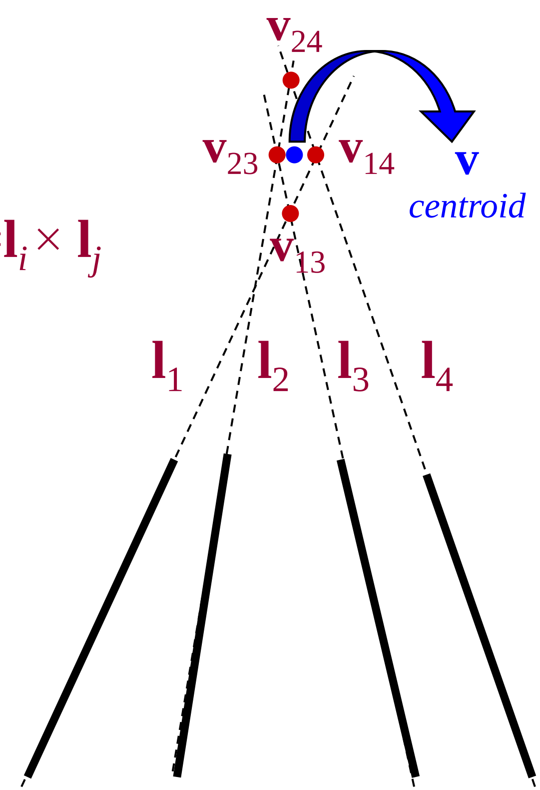
Note that **v** depends only on line direction ***d***, not on **A.**



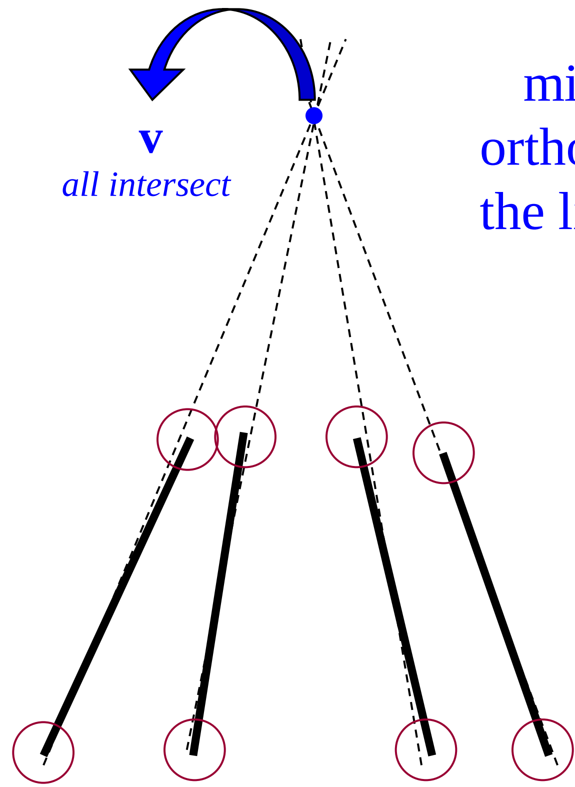
The vanishing point is the intersection of a line parallel to the scene line through the camera centre and the image plane!

On computing VPs

$\mathbf{v}_{ij}=\mathbf{l}_i \times \mathbf{l}_j$



centroid estimate



maximum likelihood estimate

minimum average orthogonal distance to the lines → non linear optimization

Contents:

- Recalling Projective Geometry
- Vanishing Points
- Camera Calibration from VPs
- Camera from Fundamental Matrix

Camera Orientation from VPs

Result 1: Consider 2 images of a scene where

- cameras differs only in orientation ($\mathcal{R}$) and position ($\boldsymbol{t}$) $\rightarrow$ (both have the same $\mathcal{K}$).
- $\mathbf{v}_i$ and $\mathbf{v}'_i$ are corresponding VPs

The directions $\boldsymbol{d}_i$ and $\boldsymbol{d}'_i$

$$\boldsymbol{d}_i = \mathcal{K}^{-1}\mathbf{v}_i \Big/ \left\|\mathcal{K}^{-1}\mathbf{v}_i\right\| \quad \text{and} \quad \boldsymbol{d}'_i = \mathcal{K}^{-1}\mathbf{v}'_i \Big/ \left\|\mathcal{K}^{-1}\mathbf{v}'_i\right\|$$

are related by the camera rotation $\mathcal{R}$

$$\boldsymbol{d}'_i = \mathcal{R}\boldsymbol{d}_i$$

Angle between Lines from VPs

Result 2: if $\mathbf{v}_1$ and $\mathbf{v}_2$ are the vanishing points of lines $\mathbf{l}_1$ and $\mathbf{l}_2$ with directions $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$, the angle between both lines can be computed from

$$\cos\theta = \frac{\boldsymbol{d}_1^T \boldsymbol{d}_2}{\|\boldsymbol{d}_1\|\|\boldsymbol{d}_2\|} = \frac{\mathbf{v}_1^T \boldsymbol{\omega} \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \boldsymbol{\omega} \mathbf{v}_1}\sqrt{\mathbf{v}_2^T \boldsymbol{\omega} \mathbf{v}_2}}$$

where $\boldsymbol{\omega} = (\mathcal{K}\mathcal{K}^T)^{-1} = \mathcal{K}^{-T}\,\mathcal{K}^{-1}$

image of the absolute conic

$\mathcal{K}$ from VPs in a single view

Result 3 : From previous slide, if $\mathbf{v}_1$ and $\mathbf{v}_2$ are the vanishing points of 2 **orthogonal lines**, then

$$\mathbf{v}_1^T \boldsymbol{\omega} \mathbf{v}_2 = 0$$

Clearly, if $\boldsymbol{\omega}$ meets the constraint above, $k\boldsymbol{\omega}$ also does, for all $k\neq0$. Thus $\boldsymbol{\omega}$ has 8 dof.

$\mathcal{K}$ from VPs in a single view

Result 4: If it is known that

- skew angle is zero $\rightarrow \mathcal{K}_{12}=0$.
- pixels are square $\rightarrow \mathcal{K}_{11}=\mathcal{K}_{22}$

$\boldsymbol{\omega}$ takes the form

$$\boldsymbol{\omega} = \begin{bmatrix} w_1 & 0 & w_2 \\ 0 & w_1 & w_3 \\ w_2 & w_3 & w_4 \end{bmatrix}$$

By setting $w_4=1$, $\boldsymbol{\omega}$ can be estimated from three orthogonal VPs.

$\mathcal{K}$ from VPs in a single view

Result 5: If it is known that

- skew angle is zero → $\mathcal{K}_{12}=0$.
- pixels are square → $\mathcal{K}_{11}=\mathcal{K}_{22}$
- image center→ $\mathcal{K}_{13}=\mathcal{K}_{23}=0$ (by offsetting vp's coordinates)

ω takes the form

$$\boldsymbol{\omega} = \begin{bmatrix} w_1 & 0 & 0 \\ 0 & w_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha^2 & 0 & 0 \\ 0 & \alpha^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1/f^2 & 0 & 0 \\ 0 & 1/f^2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

focal length in pixels

A single pair of orthogonal VPs is enough to compute **ω**.

From $\boldsymbol{\omega}$ to $\mathcal{K}$ and $\mathcal{K}'$

By definition

$$\boldsymbol{\omega}=(\mathcal{K}\mathcal{K}^{T})^{-1}=\mathcal{K}^{-T}\mathcal{K}^{-1}$$

Applying Cholesky factorization to $\boldsymbol{\omega}$ followed by inversion yields $\mathcal{K}$.

Cholesky factorization

A $n \times n$ symmetric positive-definite matrix $\mathbf{A}$ can be decomposed into a product of

- a lower triangular matrix $\mathbf{L}$ and its transpose, i.e.,

$$\mathbf{A} = \mathbf{L}\,\mathbf{L}^T$$

or

- an upper triangular matrix $\mathbf{U}$ and its transpose, i.e.

$$\mathbf{A} = \mathbf{U}^T\,\mathbf{U}$$

Assignments

Assignment 1:

Download the MATLAB demo Image2K for the subject covered in this chapter and estimate the camera matrix $\mathcal{K}$ from 2 and 3 VPs. Are the results consistent?

Assignments

Assignment 2:

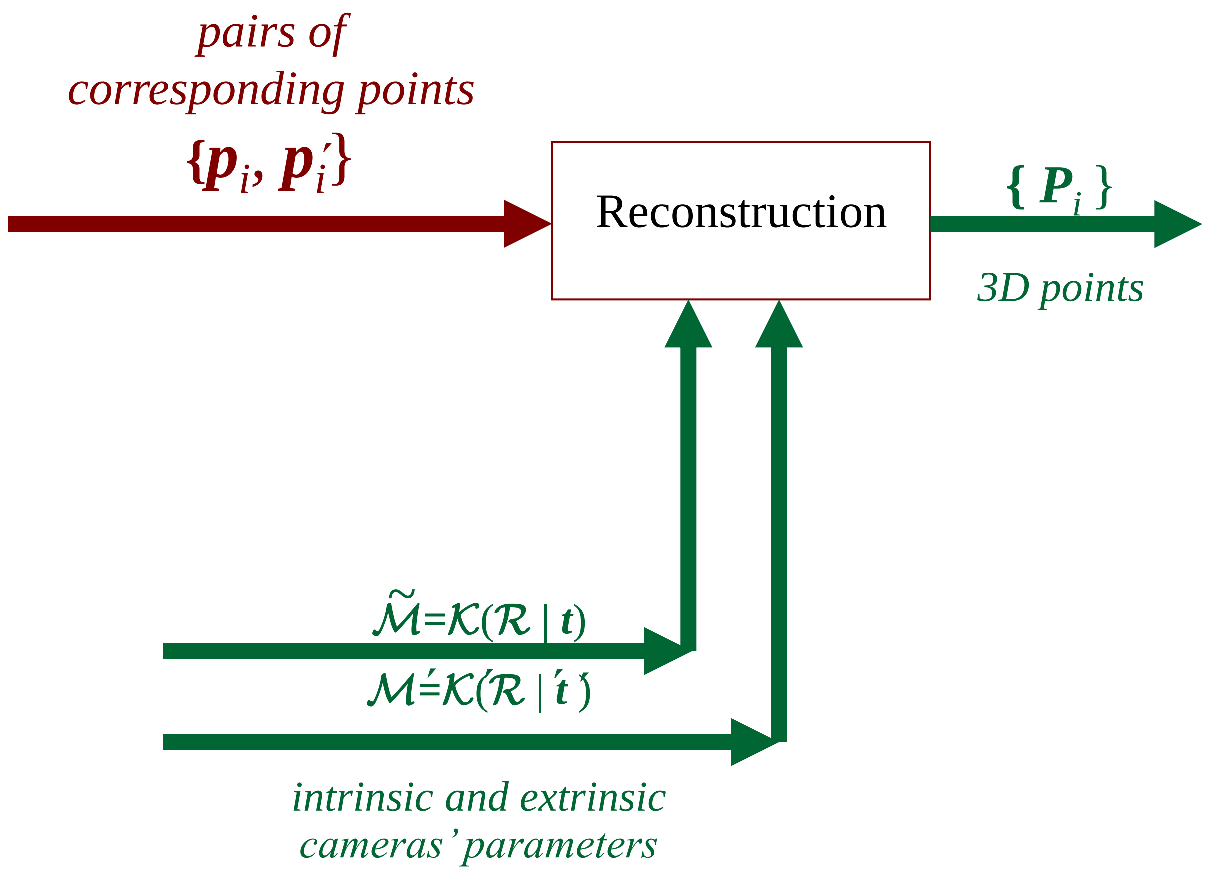
Compare the camera matrix $\mathcal{K}$ estimated by Zhang's method and by the VP based methods using the chessboard images available in Image2K.

Contents:

- Recalling Projective Geometry
- Vanishing Points
- Camera Calibration from VPs
- Camera from Fundamental Matrix

Reconstruction thus far

Require intrinsic and extrinsic cameras' parameters



Reconstruction thus far

- Reconstruction algorithms so far require knowledge of projective projection matrices

$$\mathcal{M} = \mathcal{K}(\mathcal{R} \mid \boldsymbol{t})$$
$$\mathcal{M}' = \mathcal{K}'(\mathcal{R}' \mid \boldsymbol{t}')$$

- Whereas intrinsic parameters ($\mathcal{K},\mathcal{K}'$) can be measured "once and for all", extrinsic parameters ($\mathcal{R},\boldsymbol{t},\mathcal{R}',\boldsymbol{t}'$) must be estimated for each camera positioning $\rightarrow$ troublesome!

- Next, we see a method that does not require estimating extrinsic parameters ($\mathcal{R},\boldsymbol{t},\mathcal{R}',\boldsymbol{t}'$), at least explicitly.

Recalling the essential matrix

- The extrinsic parameters ($\mathcal{R}$,$\boldsymbol{t}$,$\mathcal{R}'$,$\boldsymbol{t}'$) are embedded in the essential matrix $\mathcal{E} = \mathbf{t}\times\mathrm{R}$, where
  - $\mathrm{R}=\mathcal{R}\mathcal{R}'^{T}$ is the rotation matrix of the 2nd to 1st camera frames
  - $\mathbf{t} = \boldsymbol{t}-\mathrm{R}\boldsymbol{t}'$ is the vector defined by the origins of 1st to the 2nd camera frames
- $\mathcal{E}$ is rank 2 and has 2 equal non zero singular values.
- By setting 1st camera frame (who cares?) as the world frame ($\mathcal{R}=\mathbf{I}$, $\boldsymbol{t}=\mathbf{0}$), yields

$$(\mathcal{R},\boldsymbol{t},\mathcal{R}',\boldsymbol{t}') = (\mathbf{I}, \mathbf{0}, \mathrm{R}^{T}, -\mathrm{R}^{T}\mathbf{t})$$

Recalling the essential matrix

- In the calibrated case ($\mathcal{K}$ and $\mathcal{K}'$ known) $\mathcal{E}$ can be computed from $\mathcal{F}$

$$\mathcal{E} = \mathcal{K}'^{T} \mathcal{F} \mathcal{K}$$

- The fundamental matrix $\mathcal{F}$ can be extracted from a set of corresponding points (seen before)
- So, it might be possible to perform 3D reconstruction from $\mathcal{F}$ (and from some ground truth data).

Ambiguity given $\mathcal{F}$

Two cameras $\mathcal{M}$ and $\mathcal{M}'$ determine uniquely a fundamental matrix $\mathcal{F}$.

The converse is not true. Look

$$\boldsymbol{p} = \mathcal{M}\ \boldsymbol{P} = (\mathcal{M}\mathcal{H})\ (\mathcal{H}^{-1}\ \boldsymbol{P})$$

$$\boldsymbol{p}' = \mathcal{M}'\ \boldsymbol{P} = (\mathcal{M}'\ \mathcal{H})\ (\mathcal{H}^{-1}\ \boldsymbol{P})$$

where $\mathcal{H}$ is a 4×4 projective transformation matrix.

Thus, $\mathcal{F}$ determines the pair of camera matrices $\mathcal{M}$ and $\mathcal{M}'$ up to a 3D 4×4 projective transformation $\mathcal{H}$.

Canonical camera given $\mathcal{F}$

Let

- $\mathcal{F}$ be a fundamental matrix computed from pairs of matching points. and

- A pair of projection matrices consistent with $\mathcal{F}$ is

$$\boxed{\begin{gathered}\mathcal{M} = \mathcal{K}\,[\, \mathrm{I} \mid 0 \,] \\ \mathcal{M}' = \mathcal{K}'(\, \mathrm{R}^{T} \mid -\mathrm{R}^{T}\mathbf{t})\end{gathered}}$$

(initial estimates)

Camera from $\mathcal{E}$

Useful matrices:

$$\underbrace{\mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{orthogonal}} \quad \text{and} \quad \underbrace{\mathbf{Z} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\text{skew-symmetric}}$$

Note that $\mathbf{W}\,\mathbf{Z} = diag(1,1,0)$ and $\mathbf{Z}\,\mathbf{W}^T = -diag(1,1,0)$

Camera from $\mathcal{E}$

Result 6: If the svd of $\mathcal{E}$ is $\mathbf{U}\ \mathit{diag}(1,1,0)\mathbf{V}^T$, then

a) $[\mathbf{t}_\times]=\mathbf{UZU}^T$ and $R = \mathbf{UWV}^T$

b) $[\mathbf{t}_\times]=-\mathbf{UZU}^T$ and $R = \mathbf{UWV}^T$

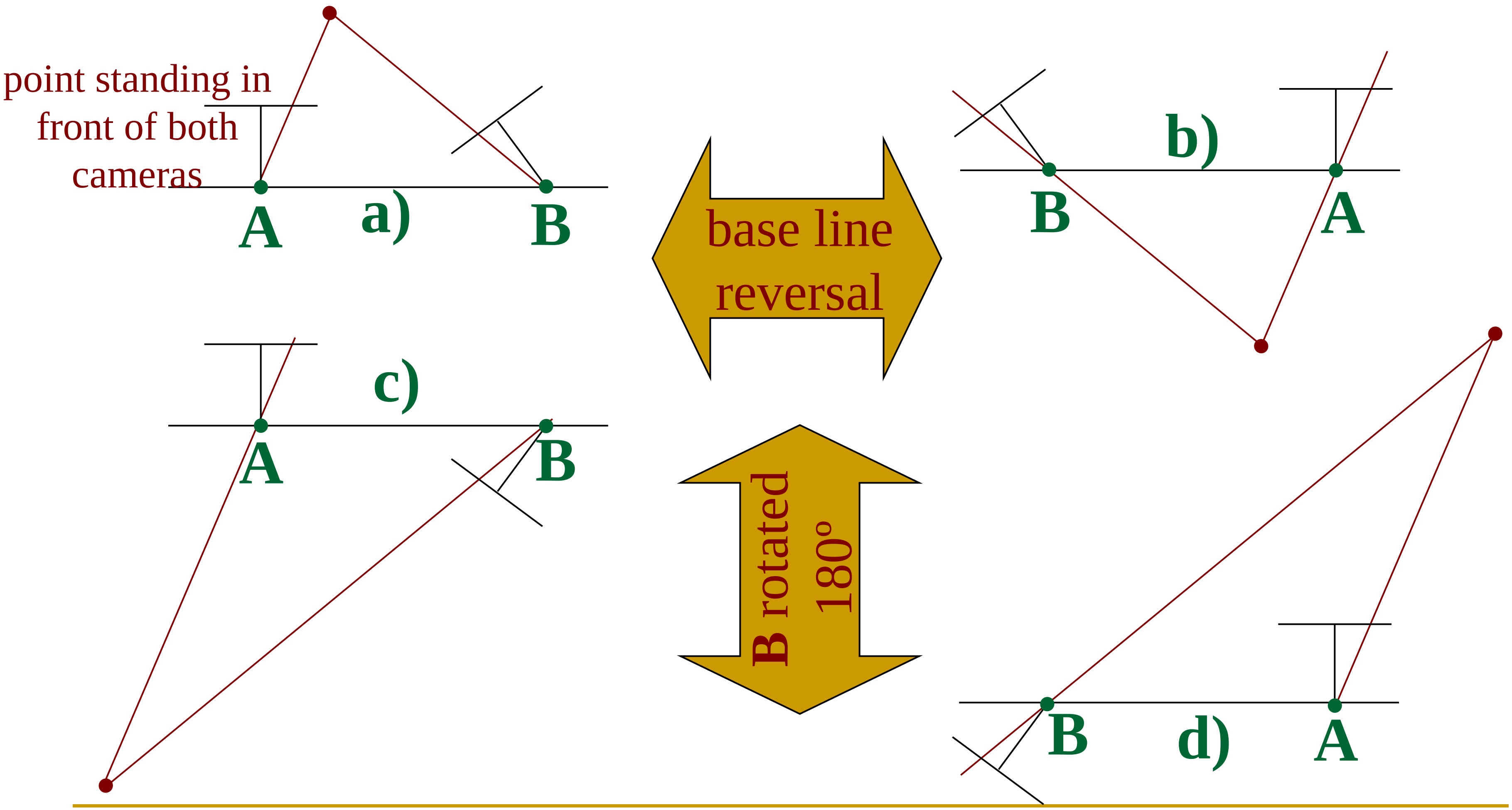
c) $[\mathbf{t}_\times]=\mathbf{UZU}^T$ and $R = \mathbf{UW}^T\mathbf{V}^T$

d) $[\mathbf{t}_\times]=-\mathbf{UZU}^T$ and $R = -\mathbf{UW}^T\mathbf{V}^T$

are possible factorizations of $\mathcal{E}$

Camera from $\mathcal{E}$

Geometric Interpretation of the 4 solutions:



Reconstruction from Ground Truth

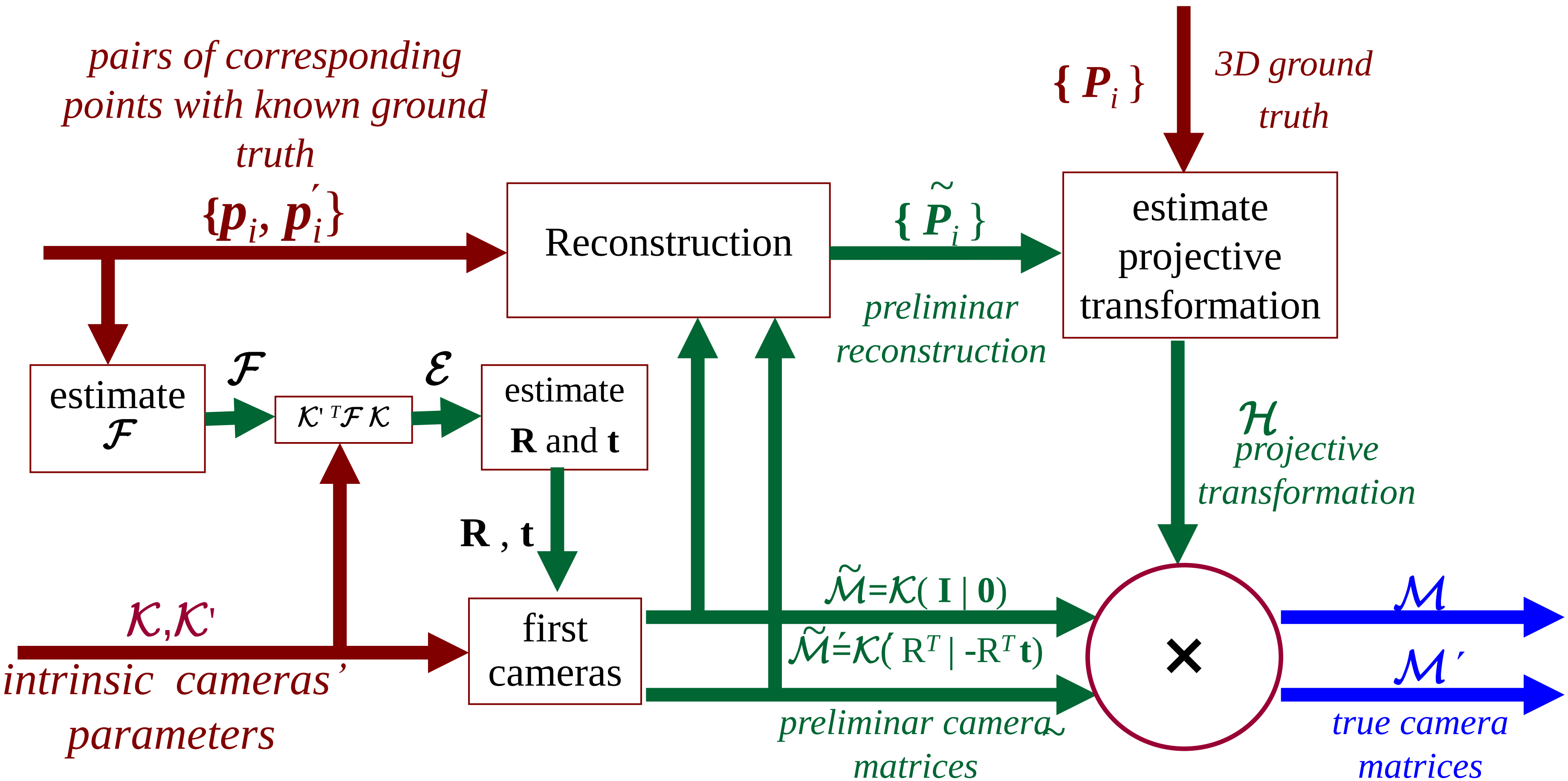
The true projective transformation $\mathcal{H}$ (or $\mathcal{H}^{-1}$) is such that

Ground truth $\Rightarrow$ $P_i = \mathcal{H}\,\tilde{P}_i$ $\Leftarrow$ measured using one pair of cameras $(\mathcal{M}, \mathcal{M}')$

- $\mathcal{H}$ has 15 dof.
- Each correspondence provides 3 independent equations.
- Thus, 5 correspondences are enough.
- Same methods for computing homographies work here.

Reconstruction from Ground Truth

Algorithm



input - intermediary result - output

Assignments

Assignment 3:

Download the MATLAB programs that implement approaches for fundamental matrix estimation. Estimate $\mathcal{F}$ for both chessboard images. Using the camera matrix $\mathcal{K}$ computed in step 1, write a program that implements the method for camera calibration based on $\mathcal{F}$ and $\mathcal{K}$ presented in the classroom.

Next Topic

Structure

from

Motion